CLAIMS

1. A data programming model, comprising:

a data item having a data item property with an associated value;

a user interface element having an element property with a value that can be defined by an association to the data item property; and

a binding definition configured to associate the element property of the user interface element with the data item property such that an application program which generates a user interface for display can be developed independent of the data item, and such that the data item can be developed independent of display-related information corresponding to the user interface element.

2. A data programming model as recited in claim 1, further comprising a transform definition configured to generate a transformed value of the data item property for association with the element property of the user interface element.

3. A data programming model as recited in claim 1, further comprising a transform definition configured to generate a transformed value of the data item property for association with the element property of the user interface element, the transformed value of the data item property being generated from the associated value of the data item property such that the associated value of the data item property is maintained unchanged in a data item database.

4. A data programming model as recited in claim 1, further comprising a transform definition developed as a logic component of the application program, the transform definition configured to generate a transformed value of the data item property for association with the element property of the user interface element.

5. A data programming model as recited in claim 1, further comprising a transform definition configured to generate a transformed value of the data item property for compatible association with the element property of the user interface element.

6. A data programming model as recited in claim 1, further comprising an update logic component configured to receive a data item update that corresponds to a change of the associated value of the data item property, and further configured to initiate that the binding definition update the element property of the user interface element with the associated value change of the data item property.

7. A data programming model as recited in claim 1, further comprising an update logic component configured to receive a user interface element update that corresponds to a change of the value of the element property of the user interface element, and further configured to update the associated data item property with the value change of the element property of the user interface element.

8. A data programming model as recited in claim 1, further comprising:

a data context property configured to define the data item as the data source of the user interface element; and

an additional binding definition configured to associate an element property of an additional user interface element with an additional data item property of the data item, the additional user interface element having a dependent association to the user interface element, and the additional binding definition further configured to default to the data context property to define the data item as the data source of the additional user interface element.

9. A data programming model as recited in claim 1, further comprising:

a data context property configured to define the data item as the data source of the user interface element;

an additional binding definition configured to associate an element property of an additional user interface element with an additional data item property of the data item, the additional user interface element having a dependent association to the user interface element, and the additional binding definition further configured to default to the data context property to define the data item as the data source of the additional user interface element; and

wherein a change of the value of the element property of the user interface element initiates a change of a value of the element property of the additional user interface element according to the default data context property.

10. A data programming model as recited in claim 1, further comprising:

a collection of data items; and

a representation of the data items each configured for display in a user interface display element that is associated with a referenced data item in the representation of the data items.

11. A data programming model as recited in claim 1, further comprising:

a collection of data items; and

a first representation of the data items in the collection and at least a second different representation of the data items in the collection, the first representation and the second different representation each being configured to reference the data items in the collection.

12. A data programming model as recited in claim 1, further comprising a data style definition configured to define a visual representation of the associated value of the data item property on the user interface.

13. A data programming model as recited in claim 1, further comprising a data style definition configured to define a template for the user interface element to display the associated value of the data item property.

14. A data programming model as recited in claim 1, further comprising a data style definition configured to define a visual representation of data items as a data tree.

15. A data programming model as recited in claim 1, further comprising a data style definition configured to define a visual representation of data items as a data tree, and wherein the data items are maintained independently from the data style definition.

16. A data programming model as recited in claim 1, further comprising:

a data style definition configured to define a visual representation of data items; and

a content presenter configured to apply the data style definition to an instantiation of a display element on the user interface to display one or more of the data items according to the defined visual representation.

17. A computing system, comprising:

an application program configured to generate a user interface having a display element to display a representation of a data item;

a data programming model configured to implement a binding definition to associate a display element property of the display element with a data item property of the data item such that a value of the data item property is displayed as the representation of the data item.

18. A computing system as recited in claim 17, wherein the data programming model includes a transform definition to generate a transformed value of the data item property for association with the display element property.

19. A computing system as recited in claim 17, wherein the data programming model includes a transform definition to generate a transformed value of the data item property for association with the display element property, the transformed value of the data item property being generated from the value of the data item property such that the value of the data item property is maintained unchanged in a data item database.

20. A computing system as recited in claim 17, wherein the application program includes a transform definition developed as a logic component of the application program, the transform definition configured to generate a transformed value of the data item property for association with the display element property.

21. A computing system as recited in claim 17, wherein the data programming model includes a transform definition configured to generate a transformed value of the data item property for compatible association with the display element property.

22. A computing system as recited in claim 17, wherein the data programming model includes an update logic component configured to receive a data item update that corresponds to a change of the value of the data item property, and is further configured to initiate that the binding definition update the display element property with the value change of the data item property.

23. A computing system as recited in claim 17, wherein the data programming model includes an update logic component configured to receive a display element update that corresponds to a change of the value of the display element property, and is further configured to update the associated data item property with the value change of the display element property.

24. A computing system as recited in claim 17, wherein the data programming model includes:

a data context property configured to define the data item as the data source of the display element; and

an additional binding definition configured to associate a display element property of an additional display element with an additional data item property of the data item, the additional display element having a dependent association to the display element, and the additional binding definition further configured to default to the data context property to define the data item as the data source of the additional display element.

25. A computing system as recited in claim 17, further comprising a collection of data items, and wherein the data programming model includes a representation of the data items each configured for display in a user interface display element that is associated with a referenced data item in the representation of the data items.

26. A computing system as recited in claim 17, further comprising a collection of data items, and wherein the data programming model includes a first representation of the data items in the collection and at least a second different representation of the data items in the collection, the first representation and the second different representation each being configured to reference the data items in the collection.

27. A computing system as recited in claim 17, wherein the data programming model includes a data style definition configured to define a visual representation of the value of the data item property on the user interface.

28. A computing system as recited in claim 17, wherein the data programming model includes a data style definition configured to define a template for the display element to display the value of the data item property.

29. A computing system as recited in claim 17, wherein the data programming model includes a data style definition configured to define a visual representation of data items as a data tree.

30. A computing system as recited in claim 17, wherein the data programming model includes a data style definition configured to define a visual representation of data items as a data tree, and wherein the data items are maintained independently from the data style definition.

31. A computing system as recited in claim 17, wherein the data programming model includes:

a data style definition configured to define a visual representation of data items; and

a content presenter configured to apply the data style definition to an instantiation of the display element on the user interface to display one or more of the data items according to the defined visual representation.

32. A method, comprising:

developing a data item having a data item property with an associated value for display on a user interface generated by an application program, the data item being developed independent of display-related information corresponding to the user interface;

developing the application program independent of the data item; and

defining an binding association between an element property of a user interface element and the data item property such that a value of the element property is defined by the association to the data item property.

33. A method as recited in claim 32, further comprising generating a transformed value of the data item property for association with the element property of the user interface element.

34. A method as recited in claim 32, further comprising generating a transformed value of the data item property for association with the element property of the user interface element, the transformed value of the data item property being generated from the associated value of the data item property such that the associated value of the data item property is maintained unchanged in a data item database.

35. A method as recited in claim 32, further comprising developing a transform definition as a logic component of the application program, the transform definition being configured to generate a transformed value of the data item property for association with the element property of the user interface element.

36. A method as recited in claim 32, further comprising generating a transformed value of the data item property for compatible association with the element property of the user interface element.

37. A method as recited in claim 32, further comprising receiving a data item update that corresponds to a change of the associated value of the data item property, and initiating that the binding definition update the element property of the user interface element with the associated value change of the data item property.

38. A method as recited in claim 32, further comprising receiving a user interface element update that corresponds to a change of the value of the element property of the user interface element, and updating the associated data item property with the value change of the element property of the user interface element.

39. A method as recited in claim 32, further comprising:

defining a data context property to identify the data item as the data source of the user interface element; and

defining an additional binding association between an element property of an additional user interface element and an additional data item property of the data item, the additional user interface element having a dependent association to the user interface element, and the additional binding association defaulting to the data context property to identify the data item as the data source of the additional user interface element.

40. A method as recited in claim 32, further comprising:

defining a collection of data items; and

developing a representation of the data items for display in user interface display elements that are each associated with a referenced data item in the representation of the data items.

41. A method as recited in claim 32, further comprising:

defining a collection of data items; and

developing a first representation of the data items in the collection and at least a second different representation of the data items in the collection, the first representation and the second different representation each referencing the data items in the collection.

42. A method as recited in claim 32, further comprising defining a data style definition for a visual representation of the associated value of the data item property on the user interface.

43. A method as recited in claim 32, further comprising defining a data style definition for a template of the user interface element to display the associated value of the data item property.

44. A method as recited in claim 32, further comprising defining a data style definition for a visual representation of data items as a data tree.

45. A method as recited in claim 32, further comprising defining a data style definition for a visual representation of data items as a data tree, the data items being maintained independently from the data style definition.

46. A method as recited in claim 32, further comprising:

defining a data style definition for a visual representation of data items; and

applying the data style definition to an instantiation of a display element on the user interface to display one or more of the data items according to the defined visual representation.

47. A method as recited in claim 32, further comprising declaring an instance of a data class which corresponds to a type of data as a resource, and wherein defining the binding association includes referring to the data class in a declaration of the binding association.